Testability 21

This chapter describes the features which are included in the embedded Pentium® processor for the purpose of enhancing testability. The capability of the Intel486™ processor test hooks are included in the embedded Pentium processor; however, some are implemented differently. In addition, new test features were added to assure timely testing and production of the system product.

Internal component testing through the Built-In Self-Test (BIST) feature provides 100% single stuck at fault coverage of the microcode ROM and large PLAs. Some testing of the instruction cache, data cache, Translation Lookaside Buffers (TLBs), and Branch Target Buffer (BTB) is also performed. In addition, the constant ROMs are checked.

Three-State Test Mode and the IEEE 1149.1 “Test Access Port and Boundary Scan” mechanism are included to facilitate testing of board connections.

See “Testability And Test Registers” on page 26-3 for more information regarding the testing of the on-chip caches, translation lookaside buffers, branch target buffer, second level caches, the superscalar architecture, and internal parity checking through the test registers.

21.1 Built-in Self-test (BIST)

Self-test is initiated by driving the INIT pin high when RESET transitions from high to low. No bus cycles are run by the embedded Pentium processor during self-test. The duration of self-test is approximately 2^{19} core clocks. Approximately 70% of the devices in the processor are tested by BIST. The BIST consists of two parts: hardware self-test and microcode self-test.

During the hardware portion of BIST, the microcode and all large PLAs are tested. All possible input combinations of the microcode ROM and PLAs are tested.

The constant ROMs, BTB, TLBs and all caches are tested by the microcode portion of BIST. The array tests (caches, TLBs and BTB) have two passes. On the first pass, data patterns are written to arrays, read back and checked for mismatches. The second pass writes the complement of the initial data pattern, reads it back and checks for mismatches. The constant ROMs are tested by using the microcode to add various constants and check the result against a stored value.

Upon completion of BIST, the cumulative result of all tests are stored in the EAX register. When EAX contains 0H, all checks passed; any non-zero result indicates a faulty unit. Note that if an internal parity error is detected during BIST, the processor will assert the IERR# pin and attempt to shutdown.

21.2 Three-state Test Mode

When the FLUSH# pin is sampled low in the clock prior to the RESET pin going from high to low, the processor enters three-state test mode. The processor floats all of its output pins and bidirectional pins including pins which are never floated during normal operation (except TDO). Three-state test mode can be initiated in order to facilitate testing of board connections. The processor remains in three-state test mode until the RESET pin is toggled again.

In a dual-processor system, the private interface pins are not floated in Three-state Test mode. These pins are PBREQ#, PBGNT#, PHIT#, and PHITM#.

Note: There are several pins that have internal pullups or pulldowns attached that show these pins going high or low, respectively, during Three-state Test mode. There is one pin, PICD1, that has an internal pulldown attached that shows this pin going low during Three-state Test mode. The five pins that have pullups are PHIT#, PHITM#, PBREQ#, PBGNT#, and PICD0. There are two other pins that have pullups attached during dual processor mode, HIT# and HITM#. The pullups on these pins (except HIT#) have a value of about 30 KOhms, HIT# is about 2 KOhms.

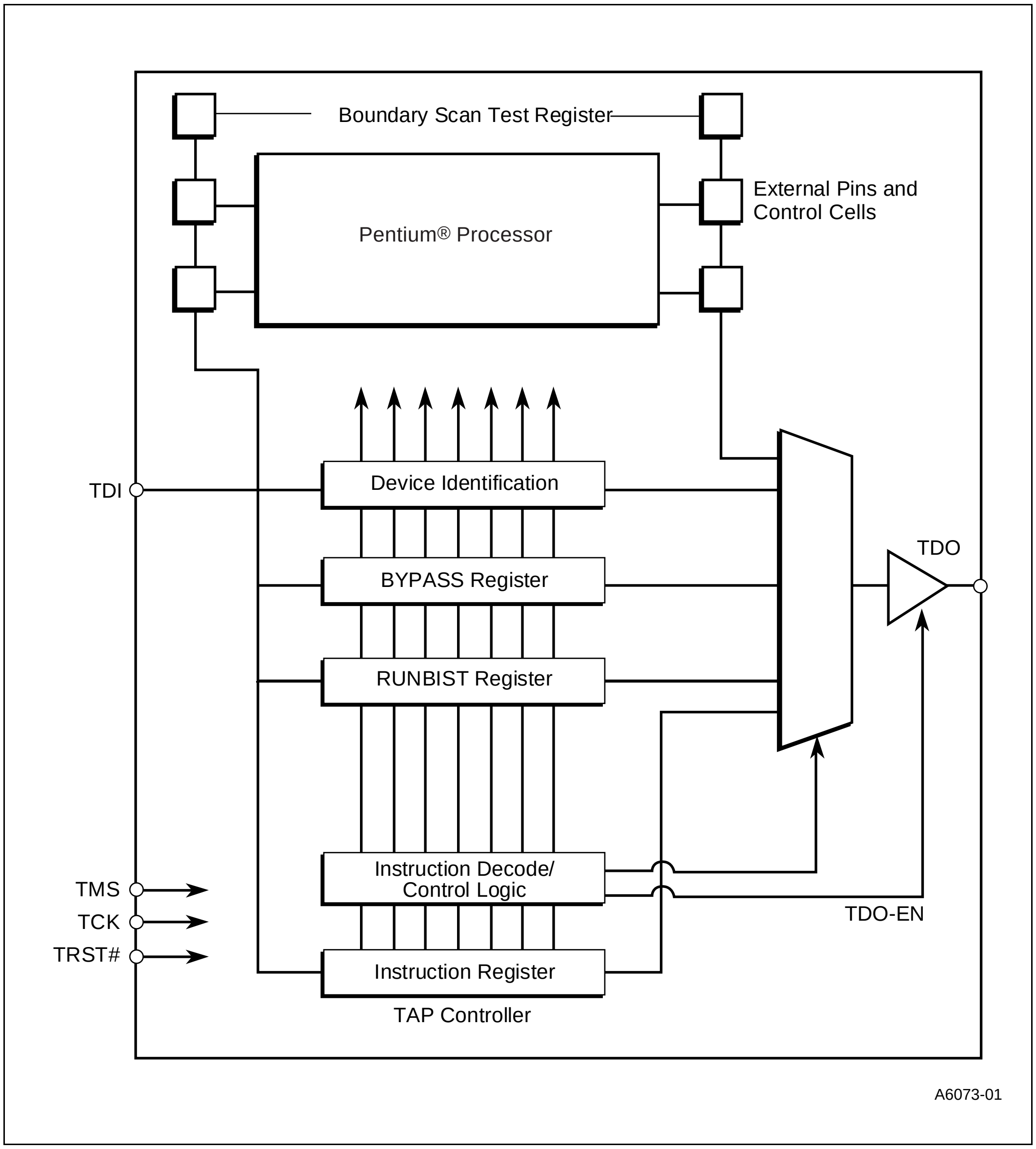
21.3 IEEE 1149.1 Test Access Port and Boundary Scan Mechanism

The IEEE Standard Test Access Port and Boundary Scan Architecture (Standard 1149.1) is implemented in the embedded Pentium processor. This feature allows board manufacturers to test board interconnects by using “boundary scan,” and to test the processor itself through BIST. All output pins are three-stateable through the IEEE 1149.1 mechanism.

21.3.1 Test Access Port (TAP)

The processor Test Access Port (TAP) contains a TAP controller, a Boundary Scan Register, four input pins (TDI, TCK, TMS, and TRST#) and one output pin (TDO). The TAP controller consists of an Instruction Register, a Device ID Register, a Bypass Register, a Runbist Register and control logic. See Figure 21-1 for the TAP Block Diagram.

Figure 21-1. Test Access Port Block Diagram



21.3.1.1 TAP Pins

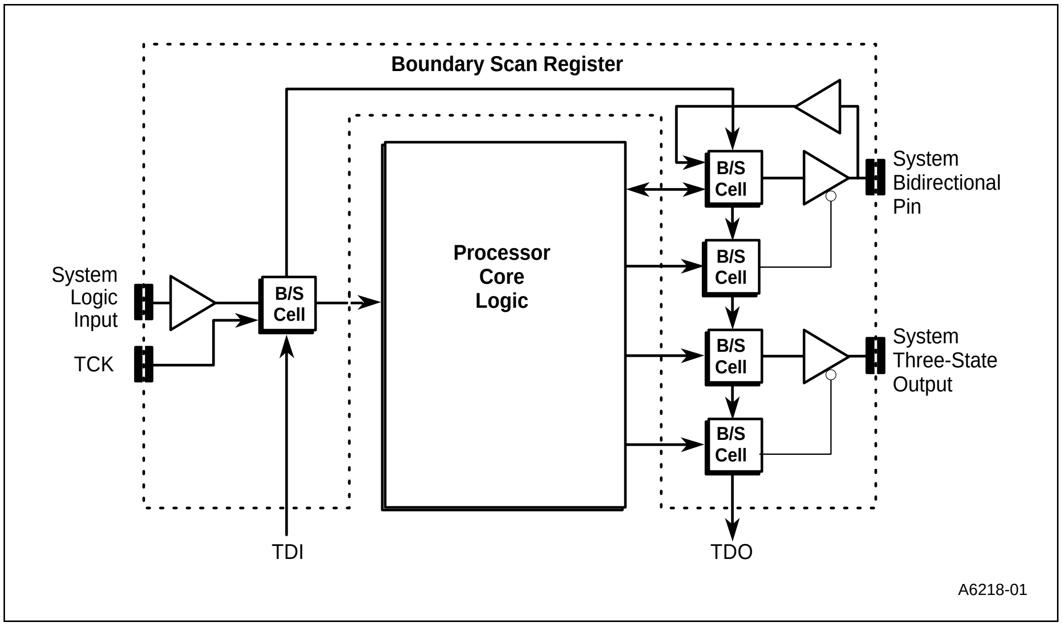
As mentioned in the previous section, the TAP includes four input pins and one output pin. TDI (test data in) is used to shift data or instructions into the TAP in a serial manner. TDO (test data out) shifts out the response data. TMS (test mode select) is used to control the state of the TAP controller. TCK is the test clock. The TDI and TMS inputs are sampled on the rising edge of this TCK. Asserting TRST# will force the TAP controller into the Test Logic Reset State (see the TAP controller state diagram, Figure 21-4). The input pins (TDI, TMS, TCK, and TRST#) have pullup resistors.

21.3.1.2 TAP Registers

Boundary Scan Register

The IEEE standard requires that an extra single bit shift register be inserted at each pin on the processor. These single bit shift registers are connected into a long shift register, the Boundary Scan Register. Therefore, the Boundary Scan Register is a single shift register path containing the boundary scan cells that are connected to all input and output pins of the processor. Figure 21-2 shows the logical structure of the Boundary Scan Register. While output cells determine the value of the signal driven on the corresponding pin, input cells only capture data; they do not affect the normal operation of the device (the INTEST instruction is not supported by the embedded Pentium processor). Data is transferred without inversion from TDI to TDO through the Boundary Scan Register during scanning. The Boundary Scan Register can be operated by the EXTEST and SAMPLE/PRELOAD instructions. The Boundary Scan Register order is defined later in this chapter.

Figure 21-2. Boundary Scan Register



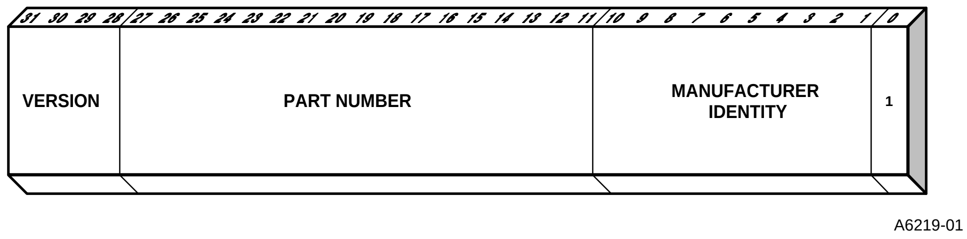
Bypass Register

The Bypass Register is a one-bit shift register that provides the minimal length path between TDI and TDO. This path can be selected when no test operation is being performed by the component to allow rapid movement of test data to and from other components on the board. While the bypass register is selected data is transferred from TDI to TDO without inversion. The Bypass Register loads a logic 0 at the start of a scan cycle.

Device ID Register

The Device Identification Register contains the manufacturer's identification code, part number code, and version code in the format shown in Figure 21-3. It is selected to be connected between TDI and TDO by using the IDCODE instruction.

Figure 21-3. Format of the Device ID Register



The processor has divided the 16-bit part number into three fields. The upper 7 bits are used to define the product type (examples: Cache, processor architecture). The middle 4 bits are used to represent the generation or family (examples: Intel486 processor, embedded Pentium processor). The lower 5 bits are used to represent the model (examples: SX, DX). Using this definition, the embedded Pentium processor ID code is shown in Table 21-1.

The version field is used to indicate the stepping ID.

Table 21-1. Device ID Register Values

Processor	Stepping	Version	Part Number			Manufacturing ID	"1"	Entire Code
			Product Type	Generation	Model			
Pentium processor (100/133/166)	x	xH	01H	05H	04H	09H	1	x82A4013H
Pentium processor with MMX™ technology	x	xH	01H	05H	03H	09H	1	x82A3013H

Runbist Register

The Runbist Register is a one bit register used to report the results of the embedded Pentium processor BIST when it is initiated by the RUNBIST instruction. This register is loaded with "0" upon successful completion of BIST.

Instruction Register

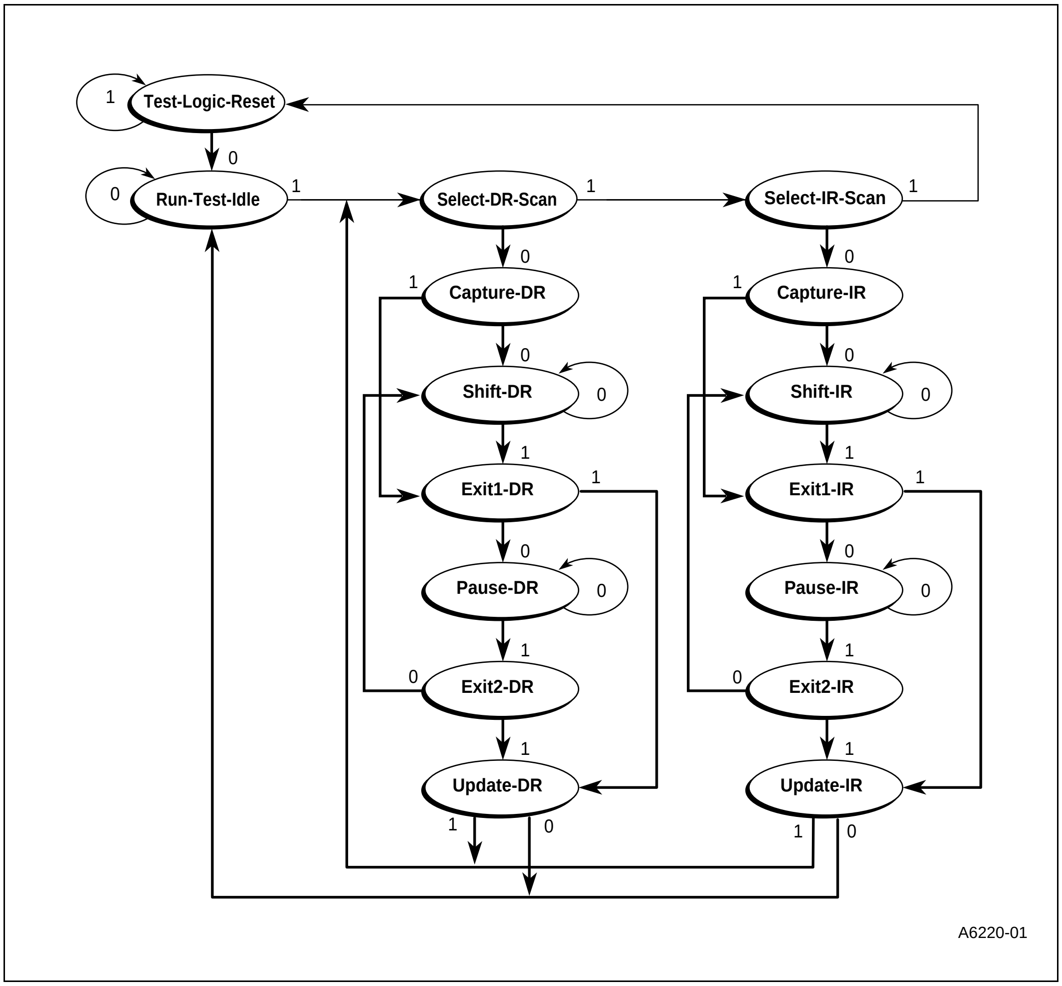
This register is 13 bits wide. The command field (the lower 4 bits of instruction) is used to indicate one of the following instructions: EXTEST, IDCODE, RUNBIST, SAMPLE/PRELOAD and BYPASS. The upper 9 bits are reserved.

The most significant bit of the Instruction Register is connected to TDI, the least significant to TDO.

21.3.1.3 TAP Controller State Diagram

Figure 21-4 shows the 16-state TAP controller state diagram. A description of each state follows. Note that the state machine contains two main branches to access either data or instruction registers.

Figure 21-4. TAP Controller State Diagram



Test-Logic-Reset State

In this state, the test logic is disabled so that normal operation of the device can continue unhindered. During initialization, the processor initializes the instruction register such that the IDCODE instruction is loaded.

No matter what the original state of the controller, the controller enters Test-Logic-Reset state when the TMS input is held high (logic 1) for at least five rising edges of TCK. The controller remains in this state while TMS is high. The TAP controller is forced to enter this state when the TRST# pin is asserted (with TCK toggling or TCK at a high logic value). The processor automatically enters this state at power-up.

Run-Test/Idle State

This is a controller state between scan operations. Once in this state, the controller remains in this state as long as TMS is held low. In devices supporting the RUNBIST instruction, the BIST is performed during this state and the result is reported in the Runbist Register. For instructions not causing functions to execute during this state, no activity occurs in the test logic. The instruction register and all test data registers retain their previous state. When TMS is high and a rising edge is applied to TCK, the controller moves to the Select-DR state.

Select-DR-Scan State

This is a temporary controller state. The test data register selected by the current instruction retains its previous state. If TMS is held low and a rising edge is applied to TCK when in this state, the controller moves into the Capture-DR state, and a scan sequence for the selected test data register is initiated. If TMS is held high and a rising edge is applied to TCK, the controller moves to the Select-IR-Scan state.

The instruction does not change in this state.

Capture-DR State

In this state, the Boundary Scan Register captures input pin data if the current instruction is EXTEST or SAMPLE/PRELOAD. The other test data registers, which do not have parallel input, are not changed.

The instruction does not change in this state.

When the TAP controller is in this state and a rising edge is applied to TCK, the controller enters the Exit1-DR state if TMS is high or the Shift-DR state if TMS is low.

Shift-DR State

In this controller state, the test data register connected between TDI and TDO as a result of the current instruction shifts data one stage toward its serial output on each rising edge of TCK.

The instruction does not change in this state.

When the TAP controller is in this state and a rising edge is applied to TCK, the controller enters the Exit1-DR state if TMS is high or remains in the Shift-DR state if TMS is low.

Exit1-DR State

This is a temporary state. While in this state, if TMS is held high, a rising edge applied to TCK causes the controller to enter the Update-DR state, which terminates the scanning process. If TMS is held low and a rising edge is applied to TCK, the controller enters the Pause-DR state.

The test data register selected by the current instruction retains its previous value during this state. The instruction does not change in this state.

Pause-DR State

The pause state allows the test controller to temporarily halt the shifting of data through the test data register in the serial path between TDI and TDO. An example use of this state could be to allow a tester to reload its pin memory from disk during application of a long test sequence.

The test data register selected by the current instruction retains its previous value during this state. The instruction does not change in this state.

The controller remains in this state as long as TMS is low. When TMS goes high and a rising edge is applied to TCK, the controller moves to the Exit2-DR state.

Exit2-DR State

This is a temporary state. While in this state, if TMS is held high, a rising edge applied to TCK causes the controller to enter the Update-DR state, which terminates the scanning process. If TMS is held low and a rising edge is applied to TCK, the controller enters the Shift-DR state.

The test data register selected by the current instruction retains its previous value during this state. The instruction does not change in this state.

Update-DR State

The Boundary Scan Register is provided with a latched parallel output to prevent changes at the parallel output while data is shifted in response to the EXTEST and SAMPLE/PRELOAD instructions. When the TAP controller is in this state and the Boundary Scan Register is selected, data is latched onto the parallel output of this register from the shift-register path on the falling edge of TCK. The data held at the latched parallel output does not change other than in this state.

All shift-register stages in the test data register selected by the current instruction retains their previous value during this state. The instruction does not change in this state.

Select-IR-Scan State

This is a temporary controller state. The test data register selected by the current instruction retains its previous state. If TMS is held low and a rising edge is applied to TCK when in this state, the controller moves into the Capture-IR state, and a scan sequence for the instruction register is initiated. If TMS is held high and a rising edge is applied to TCK, the controller moves to the Test-Logic-Reset state. The instruction does not change in this state.

Capture-IR State

In this controller state the shift register contained in the instruction register loads a fixed value on the rising edge of TCK.

The test data register selected by the current instruction retains its previous value during this state. The instruction does not change in this state.

When the controller is in this state and a rising edge is applied to TCK, the controller enters the Exit1-IR state if TMS is held high, or the Shift-IR state if TMS is held low.

Shift-IR State

In this state the shift register contained in the instruction register is connected between TDI and TDO and shifts data one stage towards its serial output on each rising edge of TCK.

The test data register selected by the current instruction retains its previous value during this state. The instruction does not change in this state.

When the controller is in this state and a rising edge is applied to TCK, the controller enters the Exit1-IR state if TMS is held high, or remains in the Shift-IR state if TMS is held low.

Exit1-IR State

This is a temporary state. While in this state, if TMS is held high, a rising edge applied to TCK causes the controller to enter the Update-IR state, which terminates the scanning process. If TMS is held low and a rising edge is applied to TCK, the controller enters the Pause-IR state.

The test data register selected by the current instruction retains its previous value during this state. The instruction does not change in this state.

Pause-IR State

The pause state allows the test controller to temporarily halt the shifting of data through the instruction register.

The test data register selected by the current instruction retains its previous value during this state. The instruction does not change in this state.

The controller remains in this state as long as TMS is low. When TMS goes high and a rising edge is applied to TCK, the controller moves to the Exit2-IR state.

Exit2-IR State

This is a temporary state. While in this state, if TMS is held high, a rising edge applied to TCK causes the controller to enter the Update-IR state, which terminates the scanning process. If TMS is held low and a rising edge is applied to TCK, the controller enters the Shift-IR state.

The test data register selected by the current instruction retains its previous value during this state. The instruction does not change in this state.

Update-IR State

The instruction shifted into the instruction register is latched onto the parallel output from the shift-register path on the falling edge of TCK. Once the new instruction has been latched, it becomes the current instruction.

Test data registers selected by the current instruction retain their previous value.

21.3.2 Boundary Scan

The IEEE Standard 1149.1 Boundary Scan is implemented using the Test Access Port and TAP Controller as described above. The embedded Pentium processor implements all of the required boundary scan features as well as some additional features. The required pins (all 3.3 V) are: TDI, TDO, TCK and TMS. The required registers are: Boundary Scan, Bypass, and the Instruction Register. Required instructions include: BYPASS, SAMPLE/PRELOAD and EXTEST. The additional pin, registers, and instructions are implemented to add additional test features.

On the board level, the TAP provides a simple serial interface that makes it possible to test all signal traces with only a few probes. The testing is controlled through the TAP Controller State machine that can be implemented with automatic test equipment or a PLD.

On power up the TAP controller is automatically initialized to the test logic reset state (test logic disabled), so normal processor behavior is the default. The Test Logic Reset State is also entered when TRST# is asserted, or when TMS is high for five or more consecutive TCK clocks.

To implement boundary scan, the TDO of one device is connected to TDI of the next in a daisy-chain fashion. This allows all of the I/O of the devices on this chain to be accessed through a long shift register. TMS and TCK are common to all devices.

The Boundary Scan Register for the embedded Pentium processor contains a cell for each pin.

The following is the bit order of the embedded Pentium processor with MMX technology Boundary Scan Register (left to right, top to bottom):

TDO→Disapsba†, PICD1, PICD0, Reserved, PICCLK, D0, D1, D2, D3, D4, D5, D6, D7, DP0, D8, D9, D10, D11, D12, D13, D14, D15, DP1, D16, D17, D18, D19, D20, D21, D22, D23, DP2, D24, D25, D26, D27, D28, D29, D30, D31, DP3, D32, D33, D34, D35, D36, D37, D38, D39, DP4, D40, D41, D42, D43, D44, D45, D46, Diswr†, D47, DP5, D48, D49, D50, D51, D52, D53, D54, D55, DP6, D56, D57, D58, D59, D60, D61, D62, D63, DP7, IERR#, FERR#, PM0BP0, PM1BP1, BP2, BP3, M/IO#, CACHE#, EWBE#, INV, AHOLD, KEN#, BRDYC#, BRDY#, BOFF#, NA#, Disbus†, Dismisch†, Disbusl†, Dismisc†, Disua2bus†, Disua1bus†, Dismisca†, Dismiscf†, WB/WT#, HOLD, PHITM#, PHIT#, PBREQ#, PBGNT#, SMIACT#, PRDY, PCHK#, APCHK#, BREQ, HLDA, AP, LOCK#, ADSC#, PCD, PWT, D/C#, EADS#, ADS#, HITM#, HIT#, W/R#, BUSCHK#, FLUSH#, A20M#, BE0#, BE1#, BE2#, BE3#, BE4#, BE5#, BE6#, BE7#, SCYC, CLK, RESET, Disabus†, A20, A19, A18, A17, A16, A15, A14, A13, A12, A11, A10, A9, A8, A7, A6, A5, A4, A3, A31, A30, A29, A28, A27, A26, A25, A24, A23, A22, A21, D/P#, NMI, RS#, INTR, SMI#, IGNNE#, INIT, PEN#, FRCMC#, Reserved, Reserved, Reserved, Reserved, Reserved, STPCLK#, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved →TDI

The following is the bit order of the embedded Pentium processor (100/133/166) Boundary Scan Register (left to right, top to bottom):

TDI → Disapsba†, PICD1, PICD0, Reserved, PICCLK, D0, D1, D2, D3, D4, D5, D6, D7, DP0, D8, D9, D10, D11, D12, D13, D14, D15, DP1, D16, D17, D18, D19, D20, D21, D22, D23, DP2, D24, D25, D26, D27, D28, D29, D30, D31, DP3, D32, D33, D34, D35, D36, D37, D38, D39, DP4, D40, D41, D42, D43, D44, D45, D46, Diswr†, D47, DP5, D48, D49, D50, D51, D52, D53, D54, D55, DP6, D56, D57, D58, D59, D60, D61, D62, D63, DP7, IERR#, FERR#, PM0/BP0, PM1/BP1, BP2, BP3, M/IO#, CACHE#, EWBE#, INV, AHOLD, KEN#, BRDYC#, BRDY#, BOFF#, NA#, Disbus†, Dismisch†, Disbus1†, Dismisc†, Disua2bus†, Disua1bus†, Dismisca†, Dismiscfa†, WB/WT#, HOLD, PHITM#, PHIT#, PBREQ#, PBGNT#, SMIACT#, PRDY, PCHK#, APCHK#, BREQ, HLDA, AP, LOCK#, ADSC#, PCD, PWT, D/C#, EADS#, ADS#, HITM#, HIT#, W/R#, BUSCHK#, FLUSH#, A20M#, BE0#, BE1#, BE2#, BE3#, BE4#, BE5#, BE6#, BE7#, SCYC, CLK, RESET, Disabus†, A20, A19, A18, A17, A16, A15, A14, A13, A12, A11, A10, A9, A8, A7, A6, A5, A4, A3, A31, A30, A29, A28, A27, A26, A25, A24, A23, A22, A21, D/P#, NMI, R/S#, INTR, SMI#, IGNNE#, INIT, PEN#, FRCMC#, Reserved, Reserved, BF0, BF1, Reserved, STPCLK#, Reserved, Reserved, Reserved, Reserved, Reserved, Reserved, CPUTYP → TDO

"Reserved" includes the no connect "NC" signals on the embedded Pentium processor.

The cells marked with an "†" are control cells that are used to select the direction of bidirectional pins or three-state the output pins. If "1" is loaded into the control cell, the associated pin(s) are three-stated or selected as input. The following lists the control cells and their corresponding pins:

For the embedded Pentium processor with MMX technology:

Disabus:	A31–A3, AP.
Disbus:	BE7–BE0#, CACHE#, SCYC, M/IO#, D/C#, W/R#, PWT, PCD.
Disbusl:	ADS#, ADSC#, LOCK#.
Dismisc:	APCHK#, PCHK#, PRDY, BP3, BP2, PM1/BP1, PM0/BP0.
Dismiscf:	D/P#.
Dismisch:	FERR#, SMIACT#, BREQ, HLDA, HIT#, HITM#.
Dismisca:	IERR#.
Disua1bus:	PBREQ#, PHIT#, PHITM#.
Disua2bus:	PBGNT#.
Diswr:	D63–D0, DP7–DP0.
Disapsba:	PICD1–PICD0

For the embedded Pentium processor (at 100/133/166 MHz):

Disabus:	A31–A3, AP
Dismiscfa:	D/P#, FERR#
Dismisca:	IERR#
Disua1buS:	PBREQ#, PHIT#, PHITM#
Disua2bus:	PBGNT#
Dismisc:	APCHK#, PHCK#, PRDY#, BP3, BP2, PM1/BP1, PM0/BP0
Disbus1:	ADS#, ADSC#, LOCK#
Dismisch:	HIT#, HITM#, HLDA, BREQ#, SMIACT#
Disbus:	SCYC, BE7#–BE0#, W/R#, D/C#, PWT, PCD, CACHE#, M/IO#
Diswr:	DP7–DP0, D63–D0
Disapsba:	PICD0, PICD1

21.3.2.1 Boundary Scan TAP Instruction Set

Table 21-2 shows the Boundary Scan TAP instructions and their instruction register encoding. A description of each instruction follows. The IDCODE and BYPASS instructions may also be executed concurrent with processor execution. The following instructions are not affected by the assertion of RESET: EXTEST, SAMPLE/PRELOAD, BYPASS, and IDCODE.

The instructions should be scanned in to the TAP port least significant bit first (bit 0 of the TAP Command field is the first bit to be scanned in).

Table 21-2. TAP Instruction Set and Instruction Register Encoding

Instruction Name	Instruction Register [Bits 12:4]	TAP Command Field [Bits 3:0]
EXTEST	XXXXXXXXX	0000
Sample/Preload	XXXXXXXXX	0001
IDCODE	XXXXXXXXX	0010
Private Instruction	XXXXXXXXX	0011
Private Instruction	XXXXXXXXX	0100
Private Instruction	XXXXXXXXX	0101
Private Instruction	XXXXXXXXX	0110
RUNBIST	XXXXXXXXX	0111
Private Instruction	XXXXXXXXX	1000
Private Instruction	XXXXXXXXX	1001
Private Instruction	XXXXXXXXX	1010
HI-Z	XXXXXXXXX	1011
Private Instruction	XXXXXXXXX	1100
BYPASS	XXXXXXXXX	1111

The TAP Command field encodings not listed in Table 21-2 (1101, 1110) are unimplemented and will be interpreted as Bypass instructions.

EXTEST The EXTEST instruction allows testing of circuitry external to the component package, typically board interconnects. It does so by driving the values loaded into the processor's Boundary Scan Register out on the output pins corresponding to each boundary scan cell and capturing the values on the processor input pins to be loaded into their corresponding Boundary Scan Register locations. I/O pins are selected as input or output, depending on the value loaded into their control setting locations in the Boundary Scan Register. Values shifted into input latches in the Boundary Scan Register are never used by the internal logic of the processor. Note: after using the EXTEST instruction, the processor must be reset before normal (non-boundary scan) use.

SAMPLE/PRELOAD The SAMPLE/PRELOAD performs two functions. When the TAP controller is in the Capture-DR state, the SAMPLE/PRELOAD instruction allows a “snap-shot” of the normal operation of the component without interfering with that normal operation. The instruction causes Boundary Scan Register cells associated with outputs to sample the value being driven by the processor. It causes the cells associated with inputs to sample the value being driven into the processor. On both outputs and inputs the sampling occurs on the rising edge of TCK. When the TAP controller is in the Update-DR state, the SAMPLE/PRELOAD instruction preloads data to the device pins to be driven to the board by executing the EXTEST instruction. Data is preloaded to the pins from the Boundary Scan Register on the falling edge of TCK.

IDCODE The IDCODE instruction selects the device identification register to be connected to TDI and TDO. This allows the device identification code to be shifted out of the device on TDO.

RUNBIST The RUNBIST instruction selects the one (1) bit Runbist Register, loads a value of “1” into the Runbist Register, and connects it to TDO. It also initiates the built-in self test (BIST) feature of the embedded Pentium processor. After loading the RUNBIST instruction code in the instruction register, the TAP controller must be placed in the Run-Test/Idle state. BIST begins on the first rising edge of TCK after entering the Run-Test/Idle state. The TAP controller must remain in the Run-Test/Idle state until BIST is completed. It requires 2^{19} core clock cycles to complete BIST and report the result to the Runbist Register. After completing BIST, the value in the Runbist Register should be shifted out on TDO during the Shift-DR state. A value of “0” being shifted out on TDO indicates BIST successfully completed. A value of “1” indicates a failure occurred. The CLK clock must be running in order to execute RUNBIST. After executing the RUNBIST instruction, the processor must be reset prior to normal (non-boundary scan) operation.

HI-Z The TAP Hi-Z instruction causes all outputs and I/Os of the embedded Pentium processor to go to a high-impedance state (float) immediately. The Hi-Z state is terminated by either resetting the TAP with the TRST# pin, by issuing another TAP instruction, or by entering the Test_Logic_Reset state. The Hi-Z state is enabled or disabled on the first TCK clock after the TAP instruction has entered the UPDATE-IR state of the TAP control state machine. This instruction overrides all other bus cycles. Resetting the processor will not disable this instruction since CPU RESET does not reset the TAP.

BYPASS The BYPASS instruction selects the Bypass Register to be connected to TDI and TDO. This effectively bypasses the test logic on the processor by reducing the shift length of the device to one bit. Note that an open circuit fault in the board level test data path will cause the Bypass Register to be selected following an instruction scan cycle due to a pull-up resistor on the TDI input. This was implemented to prevent any unwanted interference with the proper operation of the system logic.